Report by: Anand Prashar

NLP Homework 5 – q5

Q:
Repeat the investigation you conducted in question 3, but now use dense vectors from word2vec. **Comment on whether the outputs make sense. Compare the outputs of using show nearest() on word2vec and the outputs on the sparse context vector (we suggest you use the same words you used in question 3). Which method works better on the query words you choose? Please briefly explain why one method works better than other in each case.**
Note that we use the default parameters of word2vec in gensim to get word2vec word embeddings.
Complete the code used to generate answers for this question in q5.py and upload it to Vocareum for documentation purposes; provide written responses on CrowdMark below.

A

```
Experiment_list = ['edward','school','red','saved', 'eyebrows','church']

Name => 'edward'
Common noun => 'school', 'eyebrows'
Adjective => 'red'
Verb => 'saved'

Exploring further word returned from most-similar list to 'school'=> 'church'
```

Result of show nearest on above 6 words:

```
Experiment = edward

1: william (0.91890977815)
2: charles (0.894260029998)
3: robert (0.888868626274)
4: andrew (0.882156801295)
5: stephen (0.875898512972)
6: joseph (0.873689456336)
7: richard (0.865985332904)
8: alan (0.858368595339)
9: steven (0.853430295178)
10: daniel (0.842900800519)
```

Comment:
Here, we see other names have been shown similar to our experiment word. Yes, output makes sense considering all are names and are likely to have same placement in a sentence.
BELOW IS THE OUTPUT FROM SPARSE-
1: richard (0.865773624009)
2: robert (0.860447306541)
3: andrew (0.853929024452)
4: joseph (0.848822376331)
5: william (0.839134959519)
6: stephen (0.834130613871)
7: john (0.819646643676)
8: david (0.816140938498)
9: daniel (0.810912262385)
10: peter (0.809679325607)

Comparing this, we can say, SPARSE too extracted names well with similar placements. But both results do not overlap by a lot as word2Vec consider both bag of words and skipgram models.

Experiment = school
1: schools (0.75222831832)
2: college (0.749073141248)
3: class (0.62663670792)
4: student (0.580021551051)
5: classes (0.55825512395)
6: columbia (0.546274281597)
7: teacher (0.536806366718)
8: academy (0.534093683027)
9: university (0.515488570733)
10: students (0.511512873819)

Comment:
Yes, output makes sense considering all capture words that are syntactically associated with ‘school’.
BELOW IS THE OUTPUT FROM SPARSE-
1: schools (0.741096505683)
2: college (0.716161495973)
3: line (0.694353893027)
4: church (0.692936169261)
5: practice (0.692639540488)
6: experience (0.68964261712)
7: location (0.686896635079)
8: scenes (0.684265543442)
9: standards (0.68314025111)
10: movement (0.682270236675)

Pre-trained word2Vec vectors are however doing a better job here, likely due to skip-gram contribution to vectors.

Experiment = red
1: blue (0.754803342531)
2: yellow (0.661251262181)
3: pink (0.595639985565)
4: wings (0.532730105033)
5: sox (0.518735052196)
6: green (0.508377489042)
7: ice (0.500720468675)
8: shirt (0.492445267263)
9: sky (0.488523367414)
10: jeans (0.487987930956)

Comment:
Yes, output makes *obvious* sense considering all capture words that are syntactically associated directly.
BELOW IS THE OUTPUT FROM SPARSE-
1: japanese (0.774842102043)
2: french (0.773843527982)
3: russian (0.771601727641)
4: british (0.769760881161)
5: german (0.76308587011)
6: chinese (0.753910259605)
7: u.s. (0.753802151553)
8: senate (0.746977104327)
9: final (0.742776789257)
10: second (0.741986898927)

In adjectives, word2Vec is doing a much better job at capturing the syntactical information.

Experiment = saved
1: lost (0.64271176087)
2: collected (0.611404260126)
3: loved (0.542827080579)
4: hurt (0.534472012244)
5: dropped (0.53434881898)
6: missed (0.532864040169)
7: paid (0.530495099234)
8: changed (0.520907659288)
9: removed (0.519008960992)

10: bought (0.516310653236)

Comment:
Here it seems difficult to derive an immediate sense on how most of above words are related to ‘saved’.
Comparison with SPARSE result is not insightful as we had found those results were unsatisfactory themselves.

Experiment = eyebrows
1: teeth (0.567542891006)
2: lips (0.537407191856)
3: thoughts (0.532932641217)
4: hair (0.52425542498)
5: feelings (0.510657455741)
6: friendship (0.509085781474)
7: questions (0.508218613916)
8: jokes (0.507195491161)
9: shoes (0.499161971919)
10: legs (0.498542034608)

Comment:
Yes, it makes some sense but not very clear. Logically we can come up with reasoning of association for some words, but for others like ‘jokes’ etc. it is vague.

BELOW IS THE OUTPUT FROM SPARSE-
1: teeth (0.696998316688)
2: shoes (0.696253261621)
3: friends (0.687035537156)
4: hair (0.68532575405)
5: clothes (0.681900723376)
6: goals (0.678553582538)
7: skills (0.678436322527)
8: legs (0.670437263485)
9: classes (0.666140092645)
10: shots (0.662747196322)

Comparing with Sparse, Dense results are better in sense that relatively more words seem to be associated to our experiment word.

Experiment = church
1: cemetery (0.675307895233)
2: catholic (0.633414168361)
3: roman (0.632275887372)
4: christ (0.607908689327)

```
5: ceremony (0.59194697742)
6: st. (0.554892237284)
7: christian (0.543328970411)
8: jewish (0.528580305965)
9: society (0.526149212025)
10: community (0.498702322403)
```

Comment:
Yes, the results are so clear in similarity here. They all relate in aspect of Christianity.

Results from Sparse were not bad, but this result is unbeaten here.

---

3). Which method works better on the query words you choose? Please briefly explain why one method works better than other in each case.

For Name types of words, both Word2Vec and Sparse gave comparable results with slight bias towards Word2Vec-Dense.

For common Nouns, Dense words works noticeably better. The reason for this clarity can be attributed to Skip Gram which is a good way of capturing syntactical information in context.

For Adjective, again Dense was better at feeling that word had descriptive properties. Together with Bag of Words and Skip Gram, it had more chances of knowing so.

For Verbs, though not determined with clarity in chosen experiments, Dense results had few more words that made more sense. It could be an affect of Word2Vec implementing both algorithms.